

文章编号:1001-9081(2005)02-0417-03

## 基于信息熵的最少测试费用故障诊断策略

景小宁<sup>1,2</sup>, 李全通<sup>2</sup>, 陈云翔<sup>2</sup>, 吕振中<sup>2</sup>

(1. 西北工业大学 自动控制系, 陕西 西安 710072; 2. 空军工程大学 工程学院, 陕西 西安 710038)

(jxning157@sina.com)

**摘要:**针对大系统顺序故障诊断中的测试排序问题, 介绍一种基于信息熵的最少测试费用诊断树产生算法, 这种算法比传统算法计算量少, 又可有效利用测试结果、测试费用和故障概率等信息, 该算法适合于在线和离线诊断维修过程。给出了算法的设计过程, 并以实例说明算法的有效性。

**关键词:**信息熵; 故障诊断策略; 测试费用; 诊断树; 测试排序

**中图分类号:** TP206 **文献标识码:** A

## Information entropy-based fault diagnosis strategy with least test cost

JING Xiao-ning<sup>1,2</sup>, LI Quan-tong<sup>2</sup>, CHEN Yun-xiang<sup>2</sup>, LV Zhen-zhong<sup>2</sup>

(1. Department of Automatic Control, Northwestern Polytechnical University, Xi'an Shannxi 710072, China

(2. Engineering College, Air Force Engineering University, Xi'an Shannxi 710038, China)

**Abstract:** Directed towards the test sequencing problem in the sequential fault diagnosis for large scale systems, an algorithm based on information entropy for design the fault diagnosis strategy with least test cost was presented. The algorithm requires less computation than the traditional methods, and uses the test results, test cost, and fault probabilities efficiently. This algorithm is suitable for on-line or off-line diagnosis and maintenance process. The design process of the algorithm was presented, and an example was used to illustrate the validity of the method.

**Key words:** information entropy; fault diagnosis strategy; test cost; diagnosis tree; test sequencing

### 0 引言

对于如航空、航天飞机这类复杂系统, 其寿命周期费用中的维修部分将超出原购买费用的3~10倍, 高的维修费用通常可归因于在初始设计阶段缺乏可测试性需求的考虑和维修阶段缺乏有效的测试方法<sup>[1]</sup>。可测试性设计中的一个重要问题是要构建一种测试序列, 使其可获得高的故障隔离精度并消耗低的期望测试费用和故障隔离时间。顺序测试过程中考虑测试结果、测试费用和故障概率, 在外场维修中具有重要意义。

顺序故障诊断策略, 即测试排序问题, 就是要研究一种具有最小期望费用的测试过程, 当发生故障时, 来确定在有限个可能的故障源中哪个发生了故障。这个问题可用动态规划(DP)方法或AND/OR图形搜索方法(AO\*、CF和HS)求最优解。但是, 对于大系统, 动态规划DP方法或AND/OR图形搜索方法所用的计算量是非常大的<sup>[2-4]</sup>。

本文介绍一种基于信息理论的诊断树产生算法, 这种算法可克服计算量大的问题, 又可有效利用测试结果、测试费用和故障概率等信息, 获取具有最小期望费用的测试序列。

### 1 测试排序问题描述

测试排序问题包括:

1)  $m+1$ 个系统状态的有限集,  $S = \{s_0, s_1, \dots, s_m\}$ , 其中 $s_0$ 表示系统的无故障状态,  $s_i (1 \leq i \leq m)$ 表示系统的不同故障状态;

2) 与系统状态对应的 $m+1$ 个已知概率的集合,  $p = \{p_0, p_1, \dots, p_m\}$  满足  $0 < p_i \leq 1$ , 其中 $p_i$ 表示系统处于状态 $s_i$ 的先验概率, 假设最多只有一个故障,  $\sum_{i=0}^m p_i = 1$ ;

3) 可供选择的 $n$ 个二进制测试的有限集合,  $T = \{t_1, t_2, \dots, t_n\}$ ;

4)  $n$ 个测试费用的集合,  $C = \{c_1, c_2, \dots, c_n\}$ , 其中 $c_i$ 表示使用测试 $t_i$ 时所需要的费用(如包括测试设备、测试时间、人员需求等);

5) 相关矩阵 $D = [d_{ij}]$ , 其中当测试 $t_j$ 检测故障状态 $s_i$ 时,  $d_{ij} = 1$ , 否则 $d_{ij} = 0$ ;

测试排序问题的求解就是根据先前所使用的测试及其结果来确定下一个要完成的测试, 来识别未知的系统状态, 并使期望的测试费用最小化的过程。测试序列的期望费用的计算, 是通过每个所用的测试, 将其费用与其所在过程的概率相乘, 然后对测试序列中的所有测试求其和获得。即:

$$C_E = \sum_{i=1}^n c_i \times p_i$$

当相关矩阵中的所有行不相等时, 系统状态就可以明确

收稿日期: 2004-07-13

**作者简介:** 景小宁(1965-), 女, 陕西人, 副教授, 博士, 主要研究方向: 计算机测控技术、故障诊断理论及其在航空武器系统中的应用; 李全通(1963-), 男, 陕西人, 副教授, 博士研究生, 主要研究方向: 航空发动机强度、振动与可靠性, 状态监测与故障诊断; 陈云翔(1962-), 男, 江苏人, 教授, 主要研究方向: 航空维修工程、飞行器设计; 吕振中(1957-), 男, 山西人, 教授, 主要研究方向: 仿真建模, 智能信息系统。

地被识别。这样,最优测试序列就是能使问题的所有测试过程的期望费用最小的测试序列,即:

$$\min_{\{P_i\}_{i=0}^m} J = \sum_{i=0}^m \left\{ \sum_{j=1}^{|P_i|} c_{P_i[j]} \right\} p(s_i) \quad (1)$$

其中,  $P_i$  表示用于隔离系统状态  $s_i$  的测试序列,最优化是在这种可接受的测试序列类上的最优化。这里,  $|P_i|$  表示测试序列  $P_i$  的容量。

## 2 基于信息熵的最少测试费用诊断树算法

根据信息理论中熵的定义<sup>[5]</sup>,  $S = \{s_0, s_1, \dots, s_m\}$  为系统状态的有限集,  $\{p_0, p_1, \dots, p_m\}$  为该集合上的概率分布,即发生的概率,  $0 < p_i \leq 1$ , 假设最多只有一个故障,  $\sum_{i=0}^m p_i = 1$ 。该概率分布的熵为:

$$H(p) = - \sum_{i=0}^m p_i \log_2(p_i) \quad (2)$$

这里所使用的测试为二进制测试,即每个测试的结果状态有两个:通过和失败,因此产生测试序列的诊断树就是一个二元树。二元树表示了各故障模式对系统  $m+1$  个状态最大故障信息量的贡献。当已知系统  $m+1$  个状态最大故障信息量要寻找故障模式时,这时的二元树就是一个具有  $m+1$  个叶子树的外向树,二元树实际上就是一个信息传递图。

设系统可用的测试集  $T = \{t_1, t_2, \dots, t_n\}$  中的测试  $t_j$  将系统状态划分为两个子集:测试通过子集  $S_{jp}$  和测试失败子集  $S_{jf}$ , 则信息增量为:

$$IG(S, t_j) = - \{p(S_{jp}) \log_2 p(S_{jp}) + p(S_{jf}) \log_2 p(S_{jf})\} \quad (3)$$

信息增量的大小反映出故障模式导致系统故障的程度,因此二元树中路径信息增量可用于故障模式的识别。

基于最大信息熵的诊断树的产生算法如下:

1) 从所有可用的  $n$  个测试  $\{t_1, t_2, \dots, t_n\}$  中选择一个测试  $t'$ , 将  $S$  划分为通过/失败节点;

2) 设对应测试通过/失败节点的系统状态集为  $x$ , 可选用的测试集为  $t$ , 对其中每个测试  $t_j$ , 计算测试通过和失败的概率  $p(x_{jp})$  和  $p(x_{jf})$ :

$$p(x_{jp}) = \frac{\sum_{s_i \in x} [1 - d_{ij}] p(s_i)}{\hat{p}(x)} \quad (4)$$

$$p(x_{jf}) = 1 - p(x_{jp}) \quad (5)$$

$$\hat{p}(x) = \sum_{s_i \in x} p(s_i); x = x_{jp} \cup x_{jf} \quad (6)$$

3) 对每个测试计算其信息增量:

$$IG(x, t_j) = - \{p(x_{jp}) \log_2 p(x_{jp}) + p(x_{jf}) \log_2 p(x_{jf})\} \quad (7)$$

4) 从  $x$  的可用测试  $t$  中选择一个测试  $t_k$ , 使其测试的单位费用信息增益最大:

$$k = \arg \max_j \left\{ \frac{IG(x, t_j)}{c_j} \right\} \quad (8)$$

5) 测试  $t_k$  将状态集  $x$  划分为通过和失败节点;

6) 重复 2) ~ 5), 直到从测试  $t'$  开始的诊断树建立为止, 即所得到的失败节点和通过节点均为独立的故障模式为止。

7) 重复 1) ~ 6), 直到获得所有  $n$  个测试开始的诊断树为止。

8) 计算  $n$  个测试开始的诊断树中每个诊断树的平均测试费用:

$$J = \sum_{i=0}^m \left\{ \sum_{j=1}^{|P_i|} c_{P_i[j]} \right\} p(s_i) \quad (9)$$

9)  $n$  个诊断树中选取平均测试费用最小的诊断树, 就是所求的诊断树:

$$\min_{\{P_i\}_{i=0}^m} J = \sum_{i=0}^m \left\{ \sum_{j=1}^{|P_i|} c_{P_i[j]} \right\} p(s_i) \quad (10)$$

## 3 应用举例

以表 1 所示的系统为例<sup>[6]</sup>, 系统有 5 中工作状态:  $S_0, S_1, S_2, S_3$  和  $S_4$ , 其  $S_0$  为系统正常状态,  $S_1, S_2, S_3$  和  $S_4$  为系统的四种故障状态。可用的测试有 4 个, 记为:  $t_1, t_2, t_3, t_4$ 。系统的相关矩阵、故障概率和测试费用如表 1 所示。

表 1 系统相关矩阵

故障模式	测试 $t_i$ 测试费用 $C_i$	$t_1$	$t_2$	$t_3$	$t_4$	故障概率
		10	15	20	5	
$S_0$		0	0	0	0	0.80
$S_1$		1	1	1	1	0.02
$S_2$		0	1	1	0	0.03
$S_3$		0	0	1	0	0.05
$S_4$		0	0	0	1	0.10

应用上一节所述的基于最大信息熵的诊断树的产生算法, 初始节点  $S$  包含所有可能状态:  $S_0, S_1, S_2, S_3$  和  $S_4$ 。首先, 通过所有可用的 4 个测试将  $S$  划分为通过 ( $P$ ) / 失败 ( $F$ ) 节点, 如图 1 所示。

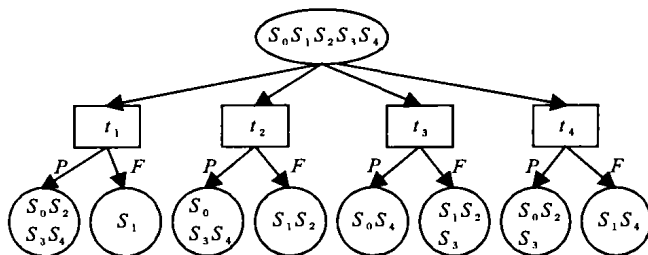


图 1 测试将系统状态划分为不同的子集节点

测试  $t_1$  将系统状态划分为两个子集:  $\{S_0, S_2, S_3, S_4\}$  和  $\{S_1\}$ , 对于子集  $\{S_0, S_2, S_3, S_4\}$  可用的测试有  $\{t_2, t_3, t_4\}$ , 根据 (4) ~ (8) 式的计算, 可得结果如表 2 所示 ( $X = \{S_0, S_2, S_3, S_4\}$ ,  $\hat{p}(x) = 0.98$ )。

根据表 2 可得所选择的测试是  $t_4$ , 测试  $t_4$  将系统状态子集  $\{S_0, S_2, S_3, S_4\}$  划分为两个子集:  $\{S_0, S_2, S_3\}$  和  $\{S_4\}$ , 对于子集  $\{S_0, S_2, S_3\}$  可用的测试有  $\{t_2, t_3\}$ , 根据 (4) ~ (8) 式的计算, 可得结果如表 3 所示 ( $X = \{S_0, S_2, S_3\}$ ,  $\hat{p}(x) = 0.88$ )。

根据表 2 可得所选择的测试是  $t_3$ , 测试  $t_3$  将系统状态子集

$\{S_0, S_2, S_3\}$  划分为两个子集:  $\{S_0\}$  和  $\{S_2, S_3\}$ , 再用测试  $t_2$  将子集  $\{S_2, S_3\}$  划分为  $\{S_3\}$  和  $\{S_2\}$ , 则得到从测试  $t_1$  开始的最大信息树。

用同样的方法, 可得到从测试  $t_2, t_3, t_4$  开始的最大信息诊断树。

表 2 子集  $\{S_0, S_2, S_3, S_4\}$  计算结果

	$t_2$	$t_3$	$t_4$
$x_p$	$\{S_0, S_3, S_4\}$	$\{S_0, S_4\}$	$\{S_0, S_2, S_3\}$
$P(x_p)$	0.95	0.90	0.88
$x_{ij}$	$\{S_2\}$	$\{S_2, S_3\}$	$\{S_4\}$
$P(x_{ij})$	0.03	0.08	0.10
$IG(x, t_j)$	0.1975	0.4079	0.4754
$IG(x, t_j)$	0.0132	0.0204	0.0951
$c_j$			
$t_k$		$t_4$	

根据式(9)计算四个诊断树的平均测试费用, 计算结果如表 4 所示。从表 4 中可以看出从测试  $t_4$  开始的最大信息诊断树就是具有最小平均测试费用的诊断树。这和文献[6]中所得结果完全相同, 但该算法的计算量明显减少。

表 3 子集  $\{S_0, S_2, S_3\}$  计算结果

	$t_2$	$t_3$
$x_p$	$\{S_0, S_3\}$	$\{S_0\}$
$P(x_p)$	0.85	0.80
$x_{ij}$	$\{S_2\}$	$\{S_2, S_3\}$
$P(x_{ij})$	0.03	0.08
$IG(x, t_j)$	0.2145	0.4395
$IG(x, t_j)$	0.0143	0.0220
$c_j$		
$t_k$		$t_3$

表 4 诊断树的平均测试费用计算结果

从 $t_i$ 开始的最大信息诊断树	要隔离的故障模式	测试序列	测试费用	诊断树的平均测试费用
$t_1$	$S_0$	$t_1, t_4, t_3$	$0.8(10 + 5 + 20) = 28$	33.7
	$S_1$	$t_1$	$0.02(10) = 0.2$	
	$S_2$	$t_1, t_4, t_3, t_2$	$0.03(10 + 5 + 20 + 15) = 1.5$	
	$S_3$	$t_1, t_4, t_3, t_2$	$0.05(10 + 5 + 20 + 15) = 2.5$	
	$S_4$	$t_1, t_4$	$0.10(10 + 5) = 1.5$	
$t_2$	$S_0$	$t_2, t_4, t_3$	$0.8(15 + 5 + 20) = 32$	37.75
	$S_1$	$t_2, t_3$	$0.02(15 + 20) = 0.7$	
	$S_2$	$t_2, t_3$	$0.03(15 + 20) = 1.05$	
	$S_3$	$t_2, t_4, t_3$	$0.05(15 + 5 + 20) = 2$	
	$S_4$	$t_2, t_4$	$0.10(15 + 5) = 2$	
$t_3$	$S_0$	$t_3, t_4$	$0.8(20 + 5) = 20$	26.2
	$S_1$	$t_3, t_4$	$0.02(20 + 5) = 0.5$	
	$S_2$	$t_3, t_4, t_2$	$0.03(20 + 5 + 15) = 1.2$	
	$S_3$	$t_3, t_4, t_2$	$0.05(20 + 5 + 15) = 2$	
	$S_4$	$t_3, t_4$	$0.10(20 + 5) = 2.5$	
$t_4$	$S_0$	$t_4, t_3$	$0.8(5 + 20) = 20$	25
	$S_1$	$t_4, t_1$	$0.02(5 + 10) = 0.3$	
	$S_2$	$t_4, t_3, t_2$	$0.03(5 + 20 + 15) = 1.2$	
	$S_3$	$t_4, t_3, t_2$	$0.05(5 + 20 + 15) = 2$	
	$S_4$	$t_4, t_1$	$0.10(5 + 10) = 1.5$	

4 结语

本文讨论的基于信息熵的最少测试费用故障诊断树的产生算法, 不用列举整个结果树的集合, 充分利用了测试结果、测试费用及故障概率信息, 减少了计算量, 适合于在线诊断测试和外场维修过程。对于所举实例, 从开始就按照单位测试费用的信息增量最大的原则选取测试, 则选择的是测试  $t_4$ , 只得到一个诊断树, 与前面结果相同, 这样算法更简单。但对于其他任意相关矩阵的情况是否使用, 还需要进一步仿真研究。

参考文献:

[1] NOVAK F, BIASIZZO A, ZELE M. Sequential Fault Diagnosis in System Maintenance [A]. Proc Of COMPEURO' 93 [C]. Paris,

1993. 119 - 124.

[2] GAREY MR. Optimal Binary Identification Procedures[J]. SIAM Journal of Applied Mathematics, 1972, 23(2): 173 - 186.

[3] BERTSEKAS DP, TSITSIKLIS JN. Neuro - Dynamic Programming [A]. VOL II, Athena Scientific[C]. Belmont, MA. 1996.

[4] PATTIPATI KR, ALEXANDRIDIS MG. Application of Heuristic Search and Information Theory to Sequential Fault Diagnosis[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1990, 20(4): 872 - 887.

[5] 徐章遂, 房立清, 王希武, 等. 故障信息诊断原理及应用[M]. 北京: 国防工业出版社, 2000.

[6] 田仲, 石君友. 系统测试性设计分析与验证[M]. 北京: 北京航空航天大学出版社, 2003.